PRÁCTICA 9

Aplicaciones multimedia

1 Objetivos

En esta práctica se pretende revisar el funcionamiento de aplicaciones multimedia como VoIP e IPTV. En particular se pretende entender el mecanismo de streaming y los diferentes protocolos que entran en juego.

2 Material

- PC Linux con software VLC, Ekiga y Wireshark.
- Auricular y micrófono.

3 Configuración del audio

Como primer paso se va a proceder a configurar el audio de su PC.

3.1 Conecte su auricular y micrófonos a los puertos adecuados de su PC. Habitualmente el color rojo de los conectores y cables ser reserva para el micrófono, y el color negro o verde para el auricular.

3.2 Abra el mezclador de audio del sistema, desde el menu "`Sistema>Preferencias>Sonido`". Si no arrancase el mezclador pruebe a lanzarlo desde la línea de comandos con el comando "`pulseaudio`". Si sigue sin aparecer pruebe el siguiente comando: "`amixer cset IFACE=mixer,name=Input Source, index=0 0`" y salte directamente al punto 3.4.

3.3 Del mezclador, seleccione la pestaña "Entrada" y sobre ella deseleccione "Silenciar". Mueva al máximo el "Volumen de entrada" y pruebe el micrófono hasta ver que se activa el indicar de niveles. Si no se activa, seleccione "Microphone 2" que corresponde al micrófono delantero del PC o "Microphone 1" hasta que compruebe que funcione el micrófono.

3.4 Arranque desde línea de comandos el programa "gnome-sound-recorder". Grabe su voz unos segundos y reproduzca el sonido para verificar que funciona correctamente la entrada y salida de audio.

4 Configuración Wireshark

Ponga el Wireshark a capturar sobre el interfaz Ethernet de su máquina para el resto de secciones siempre con las siguientes consideraciones:

4.1 Por defecto Wireshark captura sólo los 90 primeros bytes de cada paquete. Seleccione en "Capture>Interfaces>interfazencuestión>Options" el tamaño de

captura al del máximo tamaño de paquete posible en Ethernet (opción “limit each packet to”) para capturar los paquetes al completo.

4.2 Por defecto Wireshark intenta resolver las direcciones MAC (obtener el fabricante) , las direcciones IP (obtener el nombre DNS) y los puertos (obtener el nombre del servicio). Desactive el “name resolution” en “Capture>Interfaces>interfazencuestión>Options”.

5 VoIP: teleconferencia IP

Vamos a utilizar Ekiga, una aplicación de VOIP sobre SIP. Esta aplicación nos permitirá hacer y recibir llamadas VoIP desde nuestro PC, con destino dentro y fuera del laboratorio.

5.1 Ejecute desde línea de comandos el programa “`ekiga`”. Familiarícese con el interfaz. Hacer una llamada a cierto destino SIP es tan simple como introducir en la parte superior el identificador SIP del destino (sip:xxx@yyy.zzz) y pulsar sobre el icono del teléfono verde. Si le aparece un asistente de configuración seleccione el botón de cancelar.

5.2 Realice una llamada (teniendo el Wireshark capturando) al destino “sip:301@ideasip.com” que es un servicio de test de Echo que permite en primer lugar escuchar un mensaje hablado por el otro extremo y a continuación poder hablar y lo que se diga lo devuelva el otro extremo tan pronto como lo reciba. Evalúe el retardo de la voz que se produce en un sentido de la llamada. De esta forma permite evaluar el retardo que tenemos con el interlocutor. Hable durante unos segundos, finalice la llamada (pulsando en el icono del teléfono rojo) y pare la captura de Wireshark.

5.3 En Wireshark, inspeccione los paquetes capturados e identifique los protocolos de señalización y transporte utilizados en la llamada.

5.4 En cuanto al protocolo de señalización, identifique las primitivas de establecimiento y cierre de la llamada. Identifique sobre las cabeceras de esas primitivas donde se negocian los puertos que se van a utilizar posteriormente para el transporte de voz.

5.5 Determine el tamaño de los paquetes de transporte de voz ¿Por qué tienen ese tamaño grande o pequeño? ¿Cuál es el codec utilizado?

5.6 Identifique las direcciones IP que aparecen en la llamada y a qué elementos corresponden (qué función cumplen). ¿Entre cuales hay señalización o transporte de voz? ¿Por qué van a diferentes IPs?

5.7 Respecto al retardo de la voz que se produce en cada sentido de la llamada. ¿Lo puede determinar de manera objetiva? Tenga en cuenta el RTT con el interlocutor. ¿Es suficiente el retardo observado para mantener una conversación fluida?

5.8 En Wireshark, en el menú “Telephony>VoIP calls” aparece la llamada realizada. Seleccione la llamada y pulse el botón “Graph” para obtener el diagrama de

mensajes correspondiente a la conversación. Interprete el diagrama de acuerdo a lo ya evaluado en el punto 5.7.

5.9 Si el lugar del botón "Graph" pulsa el botón "Player" podría decodificar el audio de ambos sentidos de la llamada e incluso reproducirlo (funcionalidad no disponible en esta versión de Wireshark pero si en otras más recientes). Dese cuenta de que podría interceptar así cualquier llamada VoIP.

5.10 Seleccionando los paquetes del protocolo de transporte de audio en Wireshark (usando un display filter o con botón derecho sobre un paquete seleccionar "Conversation filter>UDP"), vamos a graficar el perfil de bits/sg del tráfico intercambiado mediante el menú "Statistics>IOGraphs" (selecciones Unit=bits/tick y observe la utilidad del resto de opciones). Identifique la tasa media en cada sentido ¿El codec utiliza un esquema de detección de silencios? Al cambiar en esa gráfica el "Tick interval" cambiar el perfil de la figura ¿Por qué?

5.11 En el menú "Telephony>RTP" tiene detalles de los paquetes de transporte e voz. Determine el jitter que ha experimentado en media, tiempo entre paquetes, pérdidas de paquetes, desórdenes, etc.

5.12 Realice una llamada (teniendo el Wireshark capturando) a otros compañeros de prácticas. Para ello el destino SIP será "sip:groXX@tlmYY", donde groXX es el nombre de cuenta de sus compañeros e YY es el número de máquina del laboratorio (la puede encontrar en la pegatina que tiene el PC). Conversen durante unos segundos, finalice la llamada y pare la captura de Wireshark.

5.13 Obtenga el diagrama de mensajes de la conversación e identifique los cambios con respecto a la llamada anterior.

Punto de control 9.1: Avise al profesor cuando haya completado las prácticas hasta este punto.

6 IPTV: video IP en streaming

Vamos a utilizar VLC (VideoLAN placer), una aplicación de visualización multimedia muy flexible para visualizar streaming de video por la red. En concreto, se va a utilizar para visualizar canales IPTV con contenidos en directo.

6.1 Ejecute desde línea de comandos el programa "`vlc`". Familiarícese con el interfaz.

6.2 En el laboratorio tenemos 4 canales de IPTV sobre UDP/RTP en directo en las siguientes direcciones:

- 239.0.1.1:1234: La1
- 239.0.1.2:1234: La2
- 239.0.1.3:1234: 24h
- 239.0.1.4:1234: Clan

Ponga el Wireshark a capturar. Puede visualizar un canal desde el menú de VLC "Medio>Abrir volcado de red", con protocolo=RTP, dirección=239.0.1.1 (o cualquier otra de la lista anterior), puerto=1234. Una vez aplicado, podrá visualizar el canal con calidad broadcast TV, la misma con la que lo puede visualizar en una TV vía TDT.

6.3 En Wireshark, inspeccione los paquetes capturados e identifique los protocolos de señalización y transporte utilizados. ¿La dirección 239.0.1.1 qué tipo de dirección IP es? ¿Por qué es de ese tipo?

6.4 En cuanto al protocolo de señalización, identifique las primitivas de inicio y finalización de la visualización del vídeo.

6.5 En cuanto al protocolo de transporte de audio y vídeo, Wireshark muestra que es tráfico UDP sin saber nada más. Podemos ayudarle indicándole que ese tráfico UDP en verdad es RTP. Seleccione uno de esos paquetes UDP, con botón derecho seleccione "Decode as" y elija "RTP". Ahora verá todos los paquetes de transporte como RTP. ¿Por qué ha sido necesario especificar aquí que los paquetes UDP eran RTP y en la llamada VoIP del apartado anterior no ha sido necesario?

6.6 Determine el tamaño de los paquetes de audio y video ¿Por qué tienen ese tamaño grande o pequeño? ¿Cuál es el codec utilizado?

6.7 Determine la dirección IP del servidor de vídeo que está sirviendo estos canales de IPTV.

6.8 Seleccione un paquete RTP, y con el botón derecho seleccione "Follow UDP stream". Esto une todos los datos UDP de los paquetes del mismo flujo. La mayoría son datos de audio y video (que verá como datos binarios), pero también puede encontrar cadenas de texto legibles ¿Con qué pueden estar relacionadas?

6.9 La tasa del flujo del canal de IPTV puede obtenerse desde el Wireshark menú "Statistics>UDP multicast streams". También puede obtenerlo desde el menú "Statistics>IOgraphs". Si todos sus compañeros tienen abierto el mismo canal ¿Qué tasa en bist/sg estará generando el servidor de vídeo?

6.10 En "Telephony>RTP>show all streams" obtiene el listado de conversaciones RTP capturadas, con datos generales como el jitter. Seleccione el flujo de vídeo y pulsando en el botón "Analysis" da detalle de cada paquete del flujo RTP. Pulsando "Graph" visualize los valores de jitter para cada paquete. Pulsando el botón "Save payload..." permite guardar el flujo en un fichero (por ejemplo iptv.mpg). Abrir el fichero con VLC desde el menú "Media>Abrir archivo…", y de esa manera reproducir el vídeo que se ha extraído de la captura de paquetes.

Punto de control 9.2: Avise al profesor cuando haya completado las prácticas hasta este punto.

7 YouTube: video IP en pseudostreaming

El servicio de video bajo demanda por excelencia en la actualidad es YouTube. Vamos a revisar su funcionamiento.

7.1	Realice una captura con Wireshark de la visualización de un vídeo de YouTube e idee la forma de demostrar que YouTube usa un esquema de pseudostreaming en lugar de una descarga normal vía HTTP del vídeo.
7.2	¿El reproductor está utilizando un buffer? ¿Podría estimar su tamaño?